**МИНОБРНАУКИ РОССИИ**

федеральное государственное автономное образовательное учреждение

высшего образования

«Санкт-Петербургский политехнический университет Петра Великого»

(ФГАОУ ВО «СПбПУ»)

Институт среднего профессионального образования

**ОТЧЕТ**

**по лабораторной работе №1**

по профессиональному модулю ПМ.01 «Разработка программных модулей программного обеспечения для компьютерных систем»

(код и наименование)

Специальность09.02.07 «Информационные системы и программирование»

(код и наименование специальности)

Выполнили студентки 3 курса 32919/4 группы

Кабачарова Анастасия Юрьевна, Осипова Анастасия Алексеевна

(Фамилия, имя, отчество)

Преподаватель: Ильин Ю. П.

Санкт-Петербург

2023

Содержание

**Тема: «Простейшие классы»**

**Вариант №5**

# **Цель работы:**

Изучение простейших классов, содержащих скрытые поля, конструкторы с параметрами и без параметров, методы, свойства. Рассмотреть и научиться использовать обработку исключительных ситуаций в программе.

# **Задание:**

Составить описание класса для представления комплексных чисел. Обеспечить выполнение операций сложения, вычитания и умножения комплексных чисел. Написать программу, демонстрирующую все разработанные элементы класса.

**Вариант 5**

Составить описание класса для представления комплексных чисел. Обеспечить выполнение операций сложения, вычитания и умножения комплексных чисел. Написать программу, демонстрирующую все разработанные элементы класса.

# **Математическая формулировка методов:**

1. Конструктор инициализации.

public ComplexNumber(double realPart, double imaginaryPart)//конструктор с параметрами, задающий действительную

//и мнимую части комплексного числа

{

this.realPart = realPart;

this.imaginaryPart = imaginaryPart;

}

public ComplexNumber()//конструктор, создающий комплексное число 0 + 0i

{

this.realPart = 0;

this.imaginaryPart = 0;

}

Этот код содержит два конструктора для класса ComplexNumber. Первый конструктор принимает два параметра типа double - действительную и мнимую части комплексного числа, и используется для создания объекта класса с заданными значениями этих частей. Второй конструктор не принимает аргументов и создает объект класса с действительной и мнимой частями, равными 0.

2) Метод сложения двух комплексных чисел

public ComplexNumber Add(ComplexNumber other)//метод сложения двух комплексных чисел

{

double real = this.realPart + other.realPart;//вычисление действительной части суммы

double imaginary = this.imaginaryPart + other.imaginaryPart;//вычисление мнимой части суммы

return new ComplexNumber(real, imaginary); //создание нового объекта класса ComplexNumber с полученными

//значениями

}

Данный метод описывает операцию сложения двух комплексных чисел. Он принимает в качестве параметра объект класса ComplexNumber, который представляет второе слагаемое, и вычисляет сумму действительных и мнимых частей двух комплексных чисел. Результатом является новый объект класса ComplexNumber, созданный с помощью конструктора, который принимает два аргумента - действительную и мнимую части комплексного числа.

1. Метод вычитания одного комплексного числа из другого.

public ComplexNumber Subtract(ComplexNumber other)//метод вычитания одного комплексного числа

//из другого

{

double real = this.realPart - other.realPart;//вычисление действительной части разности

double imaginary = this.imaginaryPart - other.imaginaryPart; //вычисление мнимой части разности

return new ComplexNumber(real, imaginary); //создание нового объекта класса ComplexNumber с

//полученными значениями

}

Данный метод описывает операцию вычитания двух комплексных чисел. Он принимает в качестве параметра объект класса ComplexNumber, который представляет вычитаемое число, и вычисляет разность действительных и мнимых частей двух комплексных чисел. Результатом является новый объект класса ComplexNumber, созданный с помощью конструктора, который принимает два аргумента - действительную и мнимую части комплексного числа.

1. Метод умножения двух комплексных чисел.

public ComplexNumber Multiply(ComplexNumber other)//метод умножения двух комплексных чисел

{

double real = this.realPart \* other.realPart - this.imaginaryPart \* other.imaginaryPart; //вычисление действительной части

//произведения

double imaginary = this.realPart \* other.imaginaryPart + this.imaginaryPart \* other.realPart;//вычисление мнимой части

//произведения

return new ComplexNumber(real, imaginary);//создание нового объекта класса ComplexNumber с полученными

//значениями

}

Данный метод описывает операцию умножения двух комплексных чисел. Он принимает в качестве параметра объект класса ComplexNumber, который представляет второй множитель, и вычисляет произведение двух комплексных чисел. Результатом является новый объект класса ComplexNumber, созданный с помощью конструктора, который принимает два аргумента - действительную и мнимую части комплексного числа.

1. Переопределённый метод для вывода информации об объекте.

public override string ToString()// переопределённый метод ToString(), возвращающий

//строковое представление комплексного числа

{

if (this.imaginaryPart >= 0)//если мнимая часть неотрицательна

{

return this.realPart + " + " + this.imaginaryPart + "i";//возвращаем строку вида "действительная

//часть + мнимая часть i"

}

else//если мнимая часть отрицательна

{

return this.realPart + " - " + (-this.imaginaryPart) + "i";//возвращаем строку вида "действительная

//часть - |мнимая часть| i"

}

Данный метод переопределяет стандартный метод ToString() для объектов класса ComplexNumber. Он возвращает строковое представление комплексного числа в виде "действительная часть + мнимая часть i" или "действительная часть - |мнимая часть| i", в зависимости от знака мнимой части.

**Текст программных модулей.**

//Библиотечный модуль

using System;//подключение пространства имен System

using System.Collections.Generic;//подключение пространства имен System.Collections.Generic

using System.Linq;//подключение пространства имен System.Linq

using System.Text;//подключение пространства имен System.Text

using System.Threading.Tasks; //подключение пространства имен System.Threading.Tasks

namespace ComplexNumbersLibrary//объявление пространства имен

{

public class ComplexNumber//объявление класса ComplexNumber

{

// Поля класса

private double realPart; //поле действительной части комплексного числа

private double imaginaryPart;//поле мнимой части комплексного числа

public ComplexNumber(double realPart, double imaginaryPart)//конструктор с параметрами, задающий действительную

//и мнимую части комплексного числа

{

this.realPart = realPart;

this.imaginaryPart = imaginaryPart;

}

public ComplexNumber()//конструктор, создающий комплексное число 0 + 0i

{

this.realPart = 0;

this.imaginaryPart = 0;

}

public ComplexNumber Add(ComplexNumber other)//метод сложения двух комплексных чисел

{

double real = this.realPart + other.realPart;//вычисление действительной части суммы

double imaginary = this.imaginaryPart + other.imaginaryPart;//вычисление мнимой части суммы

return new ComplexNumber(real, imaginary); //создание нового объекта класса ComplexNumber с полученными

//значениями

}

public ComplexNumber Subtract(ComplexNumber other)//метод вычитания одного комплексного числа

//из другого

{

double real = this.realPart - other.realPart;//вычисление действительной части разности

double imaginary = this.imaginaryPart - other.imaginaryPart; //вычисление мнимой части разности

return new ComplexNumber(real, imaginary); //создание нового объекта класса ComplexNumber с

//полученными значениями

}

public ComplexNumber Multiply(ComplexNumber other)//метод умножения двух комплексных чисел

{

double real = this.realPart \* other.realPart - this.imaginaryPart \* other.imaginaryPart; //вычисление действительной части

//произведения

double imaginary = this.realPart \* other.imaginaryPart + this.imaginaryPart \* other.realPart;//вычисление мнимой части

//произведения

return new ComplexNumber(real, imaginary);//создание нового объекта класса ComplexNumber с полученными

//значениями

}

public override string ToString()// метод ToString(), возвращающий

//строковое представление комплексного числа

{

if (this.imaginaryPart >= 0)//если мнимая часть неотрицательна

{

return this.realPart + " + " + this.imaginaryPart + "i";//возвращаем строку вида "действительная

//часть + мнимая часть i"

}

else//если мнимая часть отрицательна

{

return this.realPart + " - " + (-this.imaginaryPart) + "i";//возвращаем строку вида "действительная

//часть - |мнимая часть| i"

}

}

}

}

// консольное приложение

using System;//подключение пространства имен System

using ComplexNumbersLibrary;

class Program

{

static void Main(string[] args)

{

ComplexNumber z1 = new ComplexNumber(8, 9); // создание комплексного числа 2 + 3i

ComplexNumber z2 = new ComplexNumber(-2, 3); // создание комплексного числа -1 + 2i

Console.WriteLine("z1 = " + z1); // вывод на экран комплексного числа z1

Console.WriteLine("z2 = " + z2); // вывод на экран комплексного числа z2

ComplexNumber sum = z1.Add(z2); // сложение комплексных чисел z1 и z2

ComplexNumber difference = z1.Subtract(z2); // вычитание комплексного числа z2 из z1

ComplexNumber product = z1.Multiply(z2); // умножение комплексных чисел z1 и z2

Console.WriteLine("z1 + z2 = " + sum); // вывод на экран суммы комплексных чисел z1 и z2

Console.WriteLine("z1 - z2 = " + difference); // вывод на экран разности комплексных чисел z1 и z2

Console.WriteLine("z1 \* z2 = " + product); // вывод на экран произведения комплексных чисел z1 и z2

Console.ReadKey();

}

}

|  |  |  |  |
| --- | --- | --- | --- |
| **Проверяемые требования** | **Сообщения программы и вводимые значения** | **Ожидаемые результаты** | **Фактические результаты** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Протокол испытаний:**

**Результат выполнения работы программы:**

1. Первой строчкой в консольном выводе выводится значение z1= 8+9i;
2. Второй строчкой в консольном выводе выводится значение z2=-2+3i;
3. Третьей строчкой в консольном выводе выполняется операция сложения двух комплексных чисел;
4. Четвертой строчкой в консольном выводе выполняется операция вычитания одного комплексного числа из другого;
5. Пятой строчкой в консольном выводе выполняется операция умножения двух комплексных чисел.

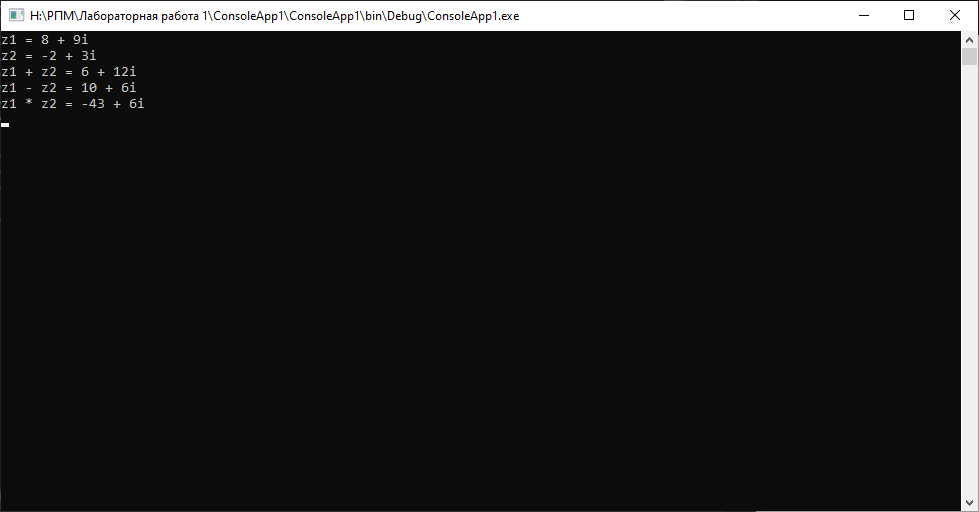


Рисунок 1 – результат выполнения задания со значениями z1= (8,9) и

z2=(-2,3).